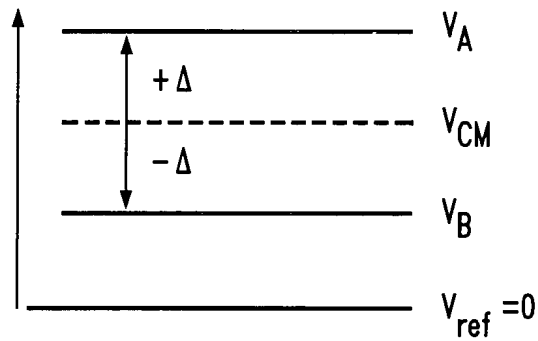


*FIG. 1*

PRIOR ART



*FIG. 2*

PRIOR ART

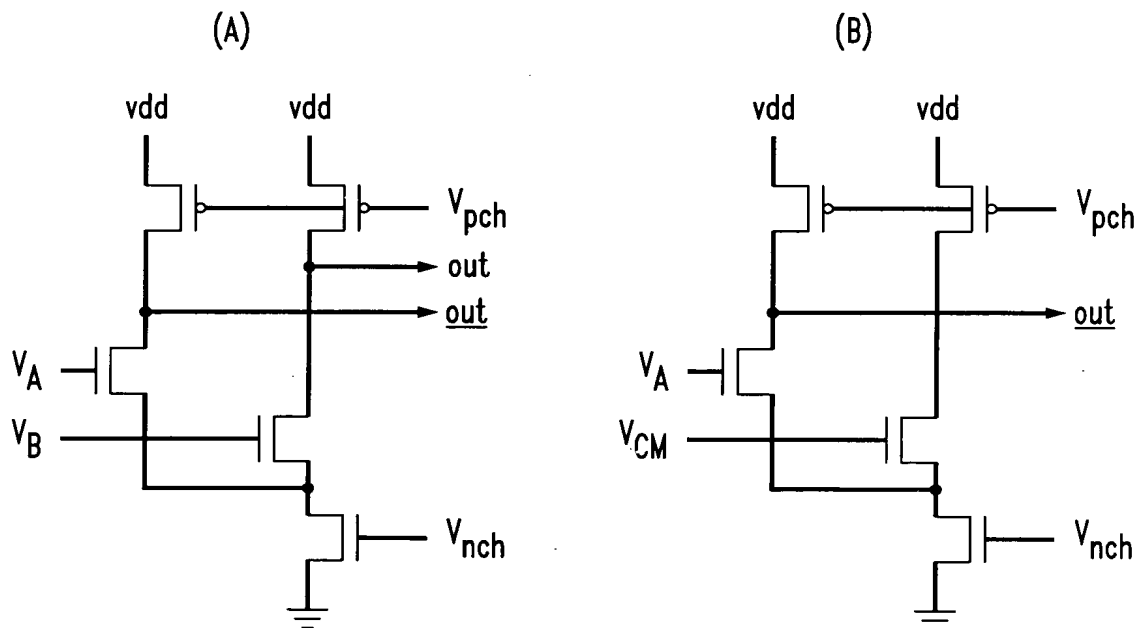


FIG. 3

PRIOR ART

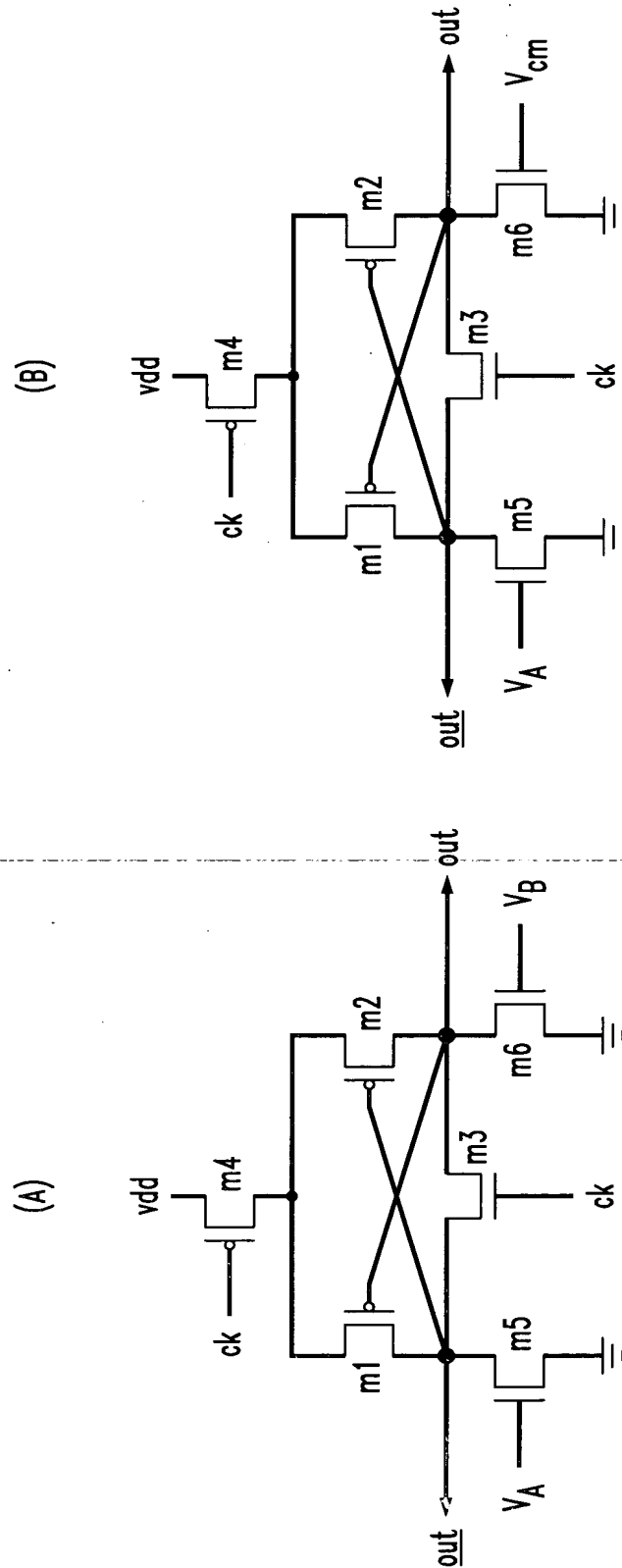
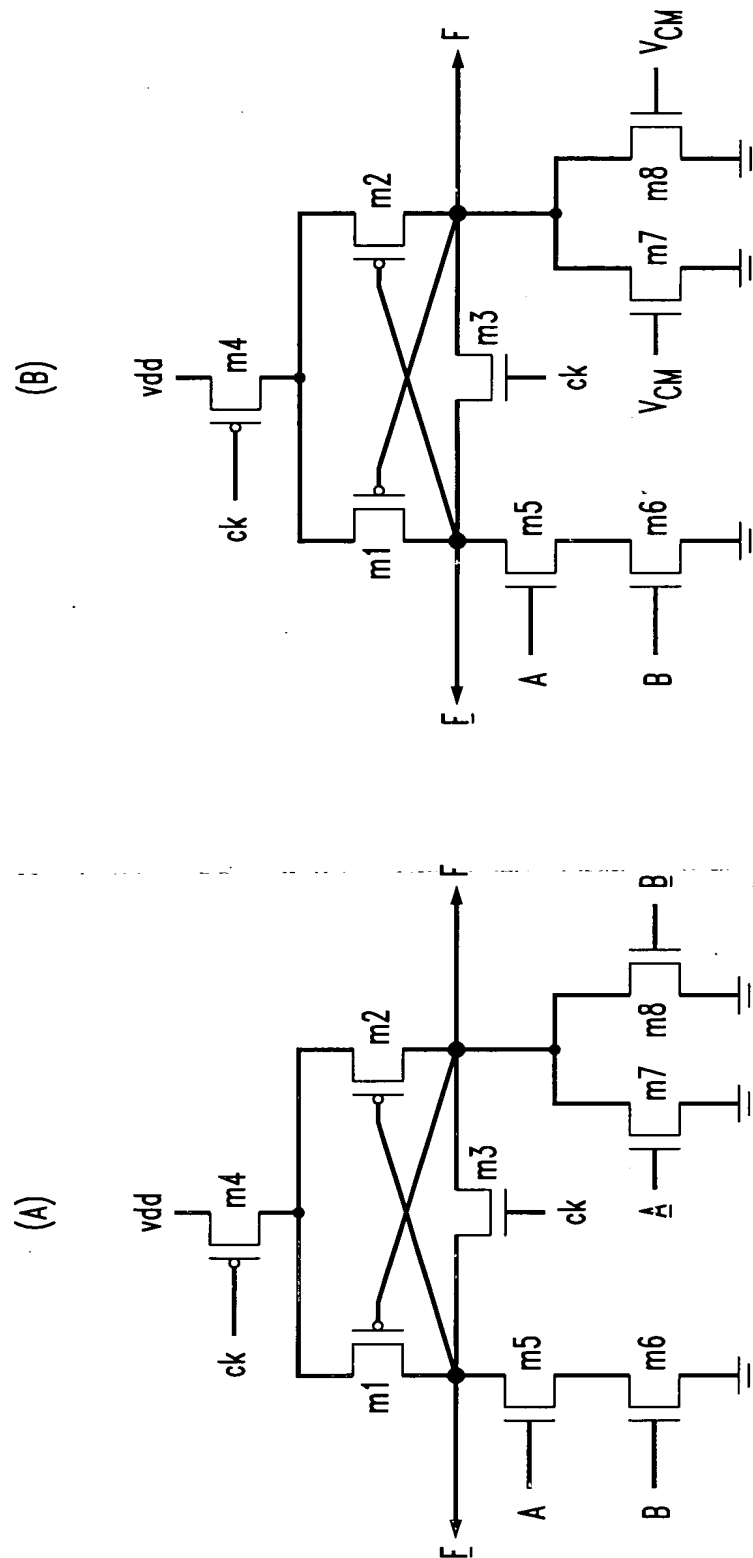


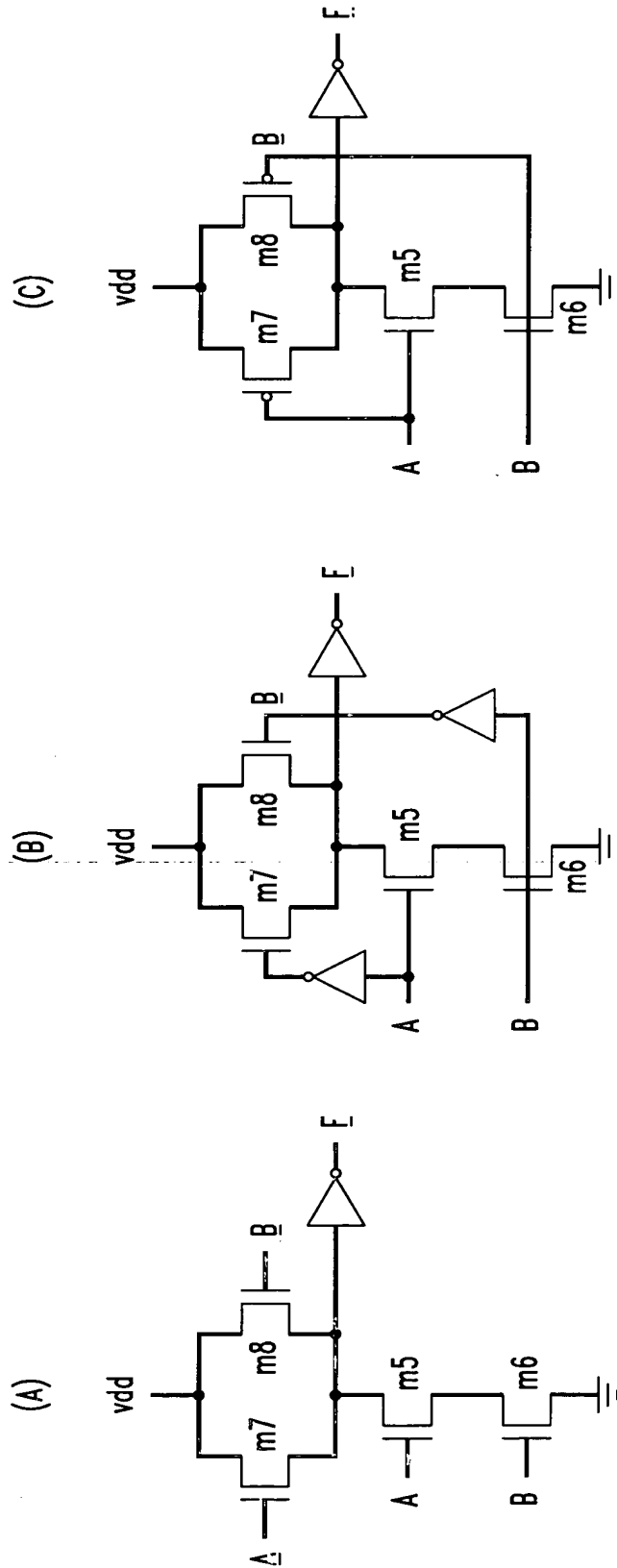
FIG. 4

PRIOR ART



4/32

FIG. 5  
PRIOR ART

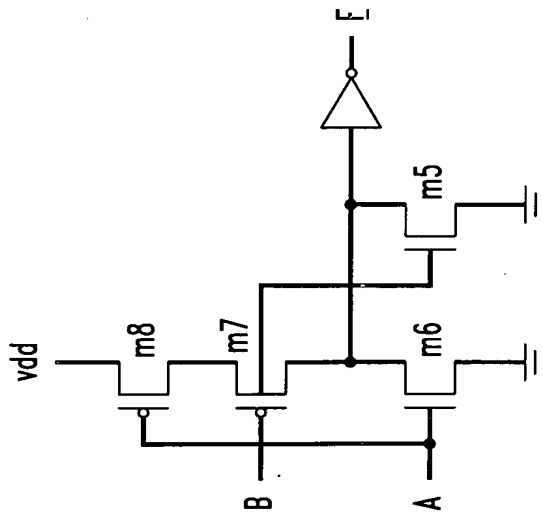


5/32

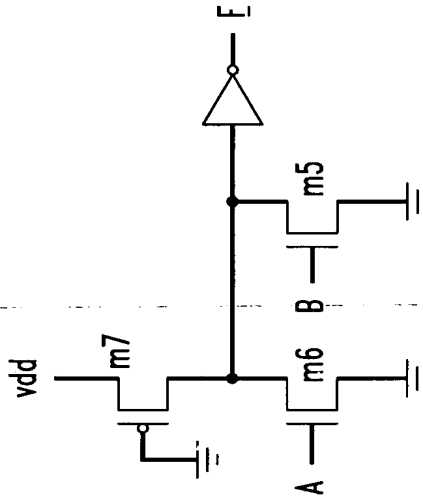
FIG. 6

PRIOR ART

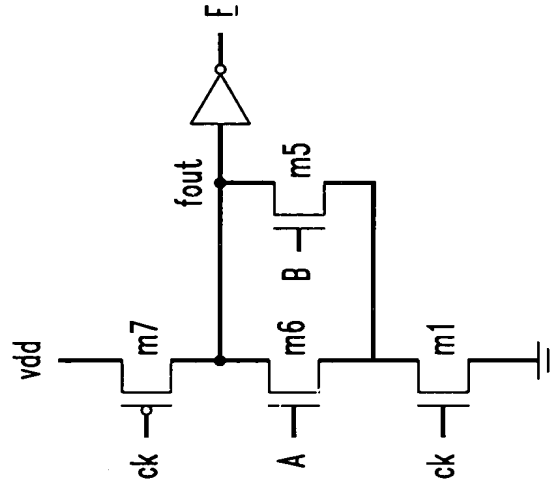
(A)



(B)



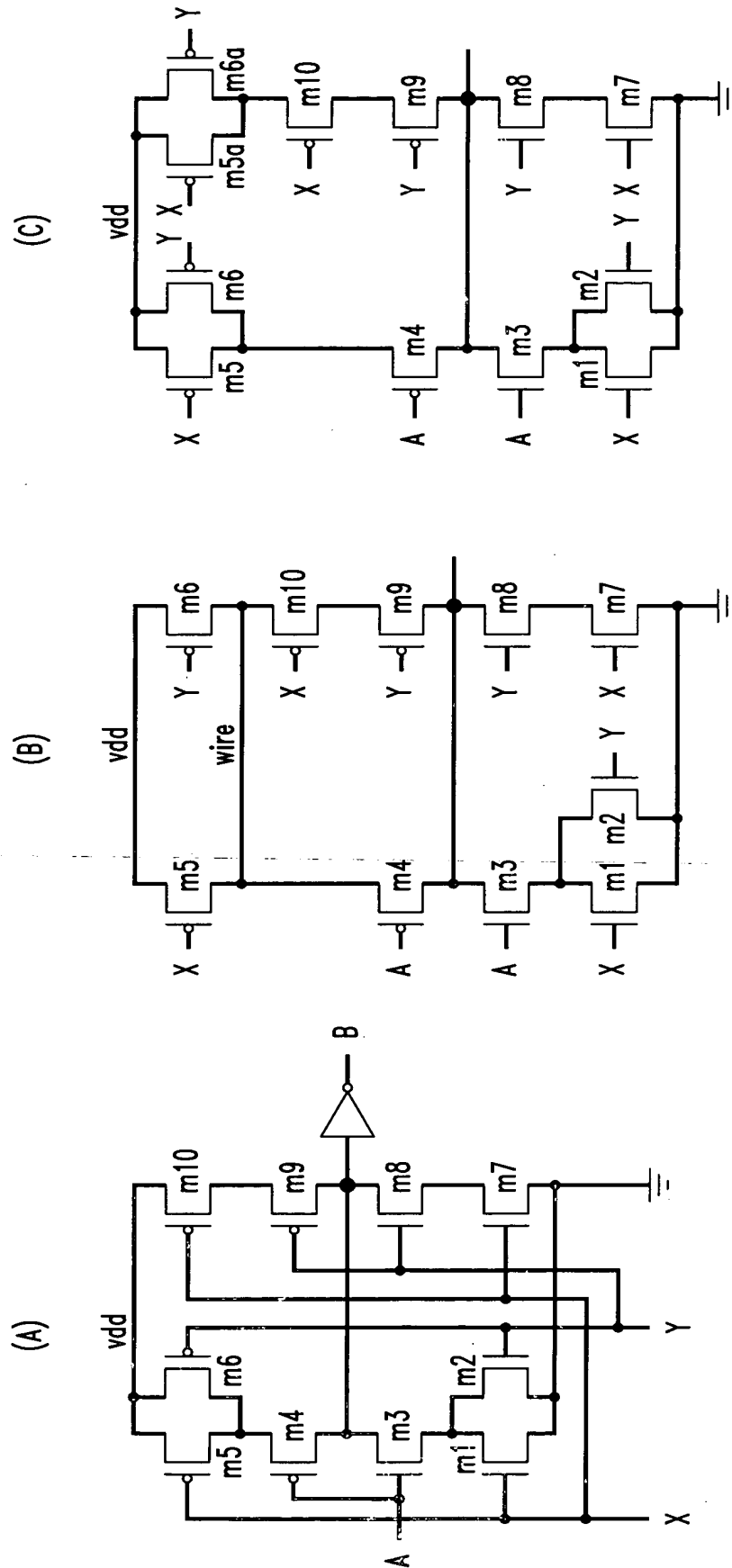
(C)



6/32

FIG. 7

PRIOR ART



7/32

FIG. 8

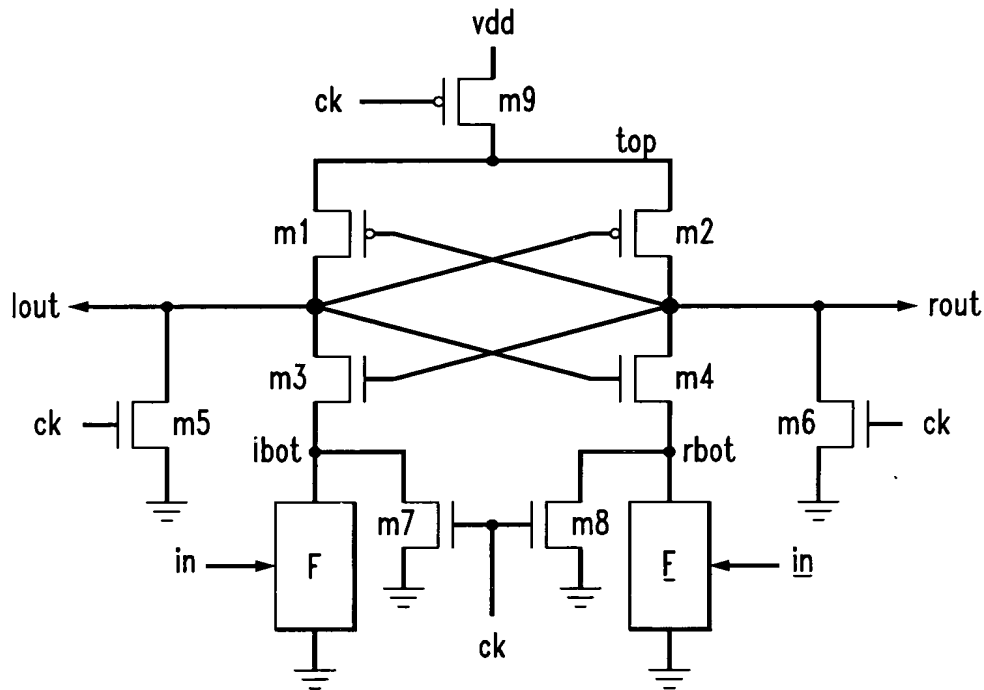
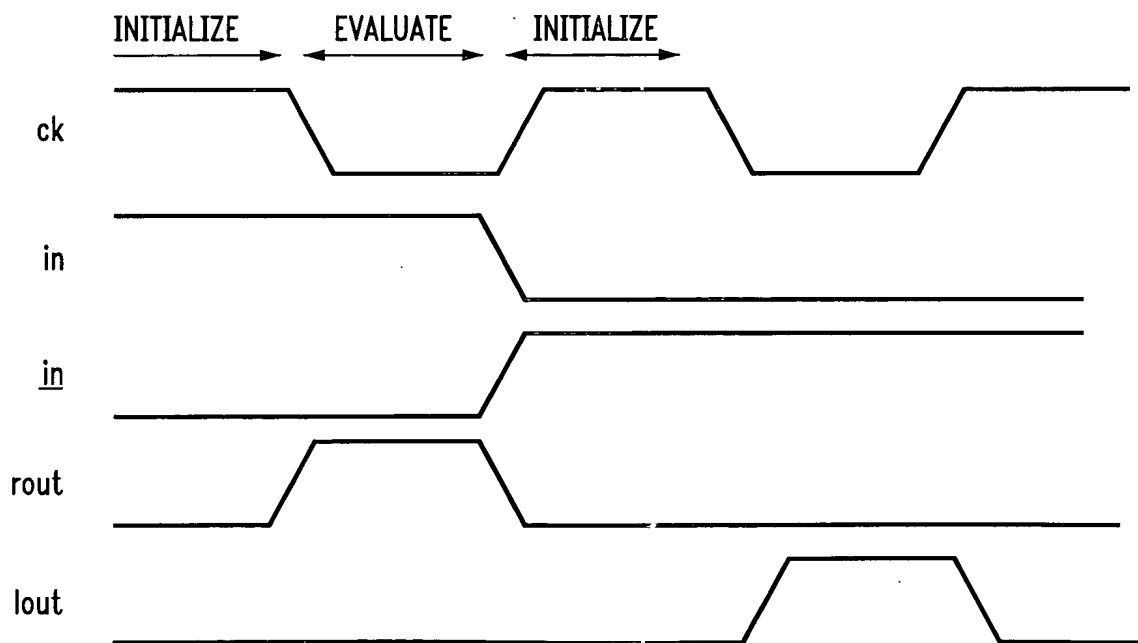


FIG. 9



8/32

FIG. 10A

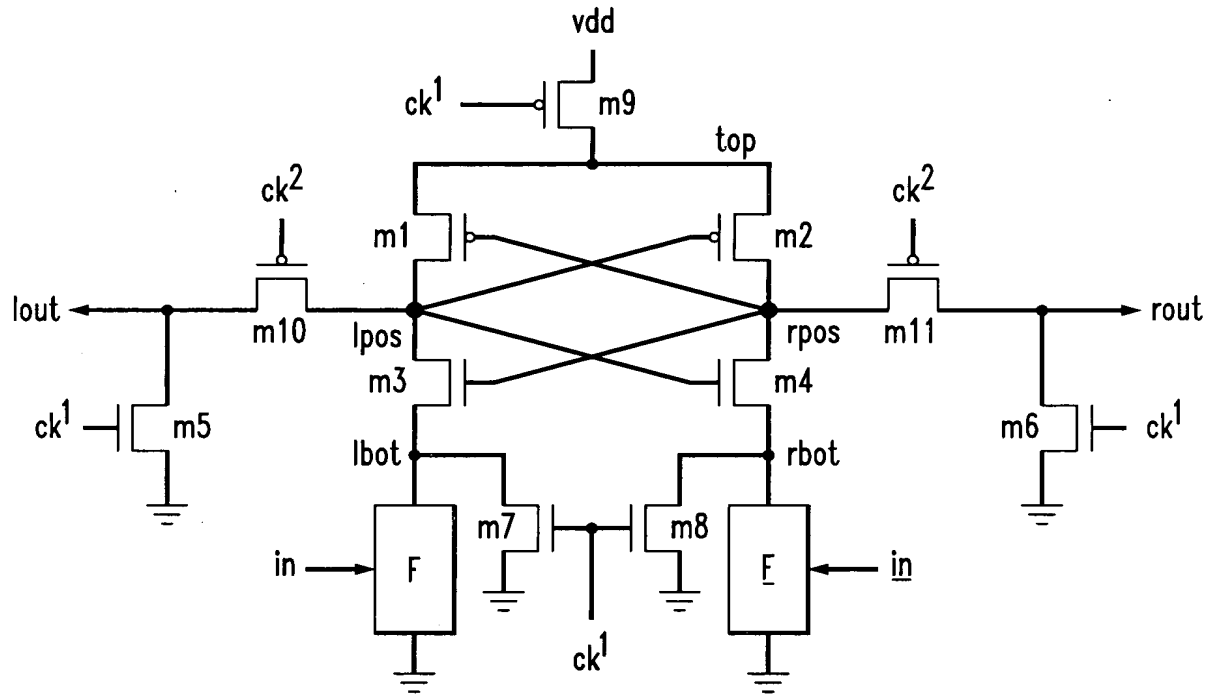
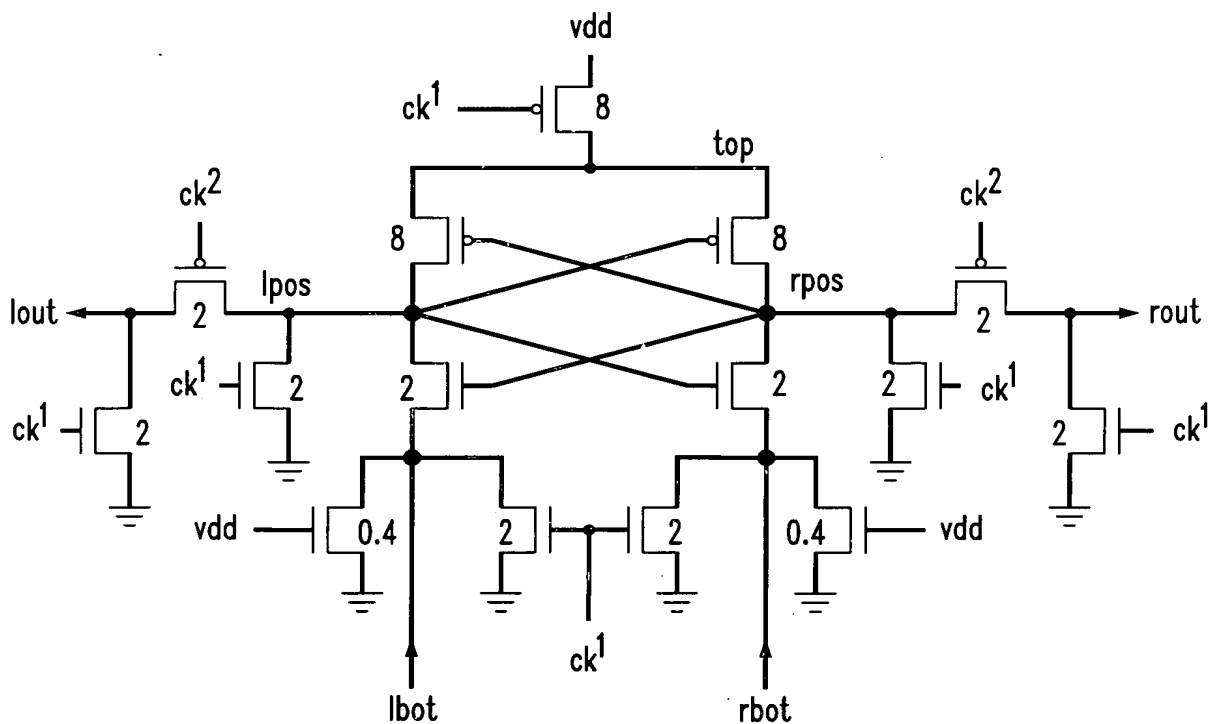


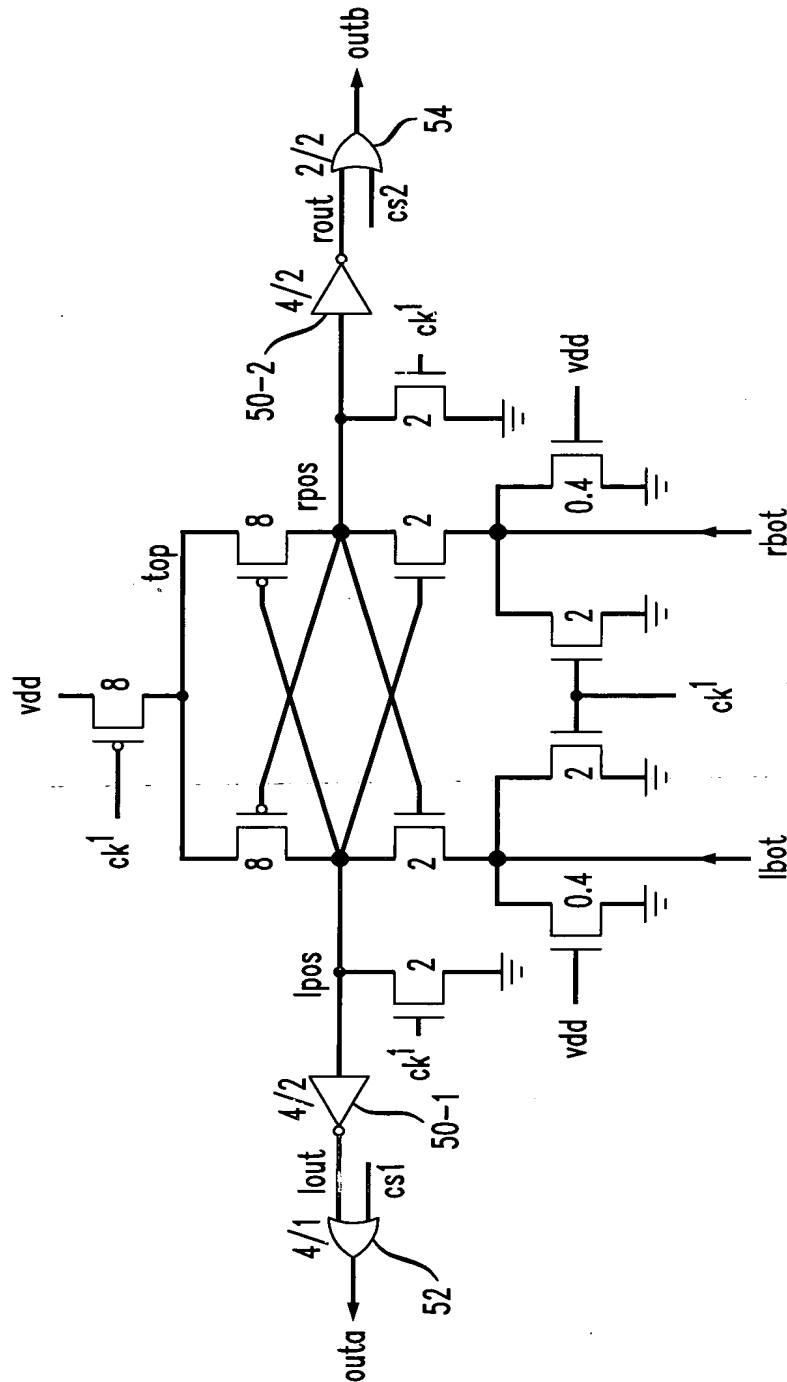
FIG. 10B





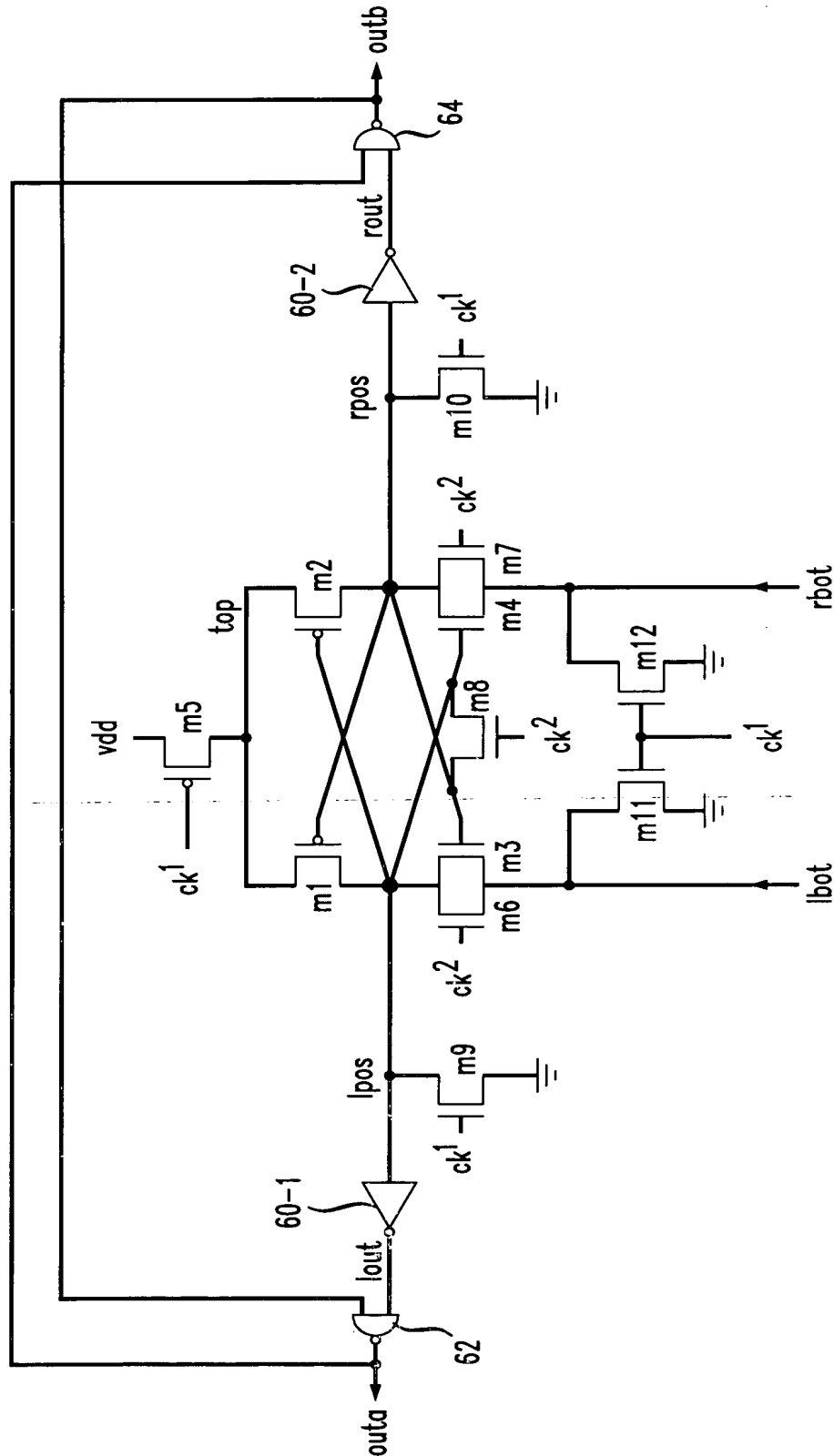
9/32

FIG. 10C



10/32

FIG. 10D



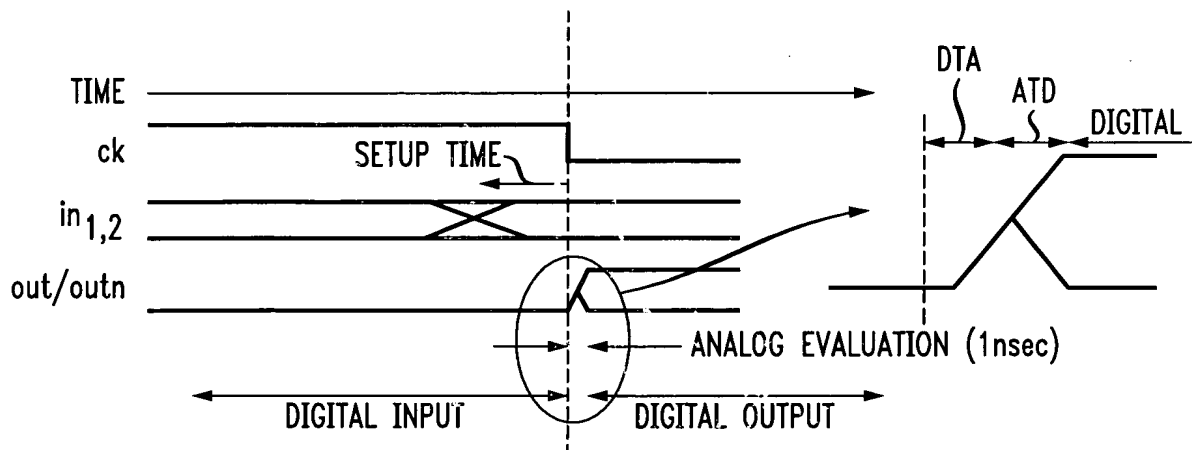
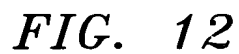


FIG. 13

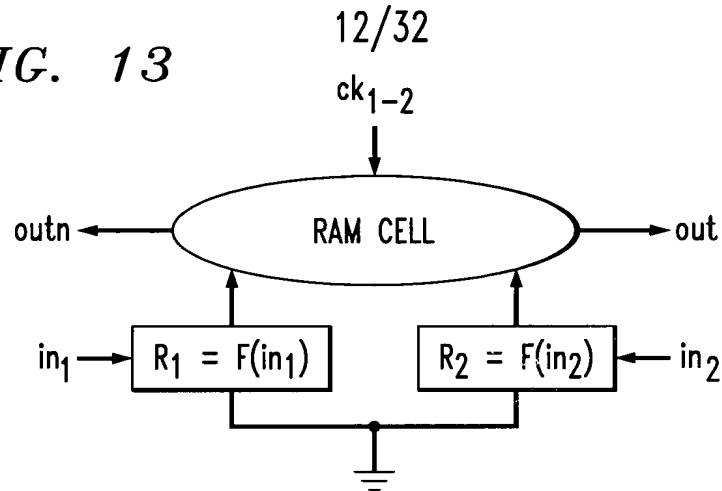


FIG. 14

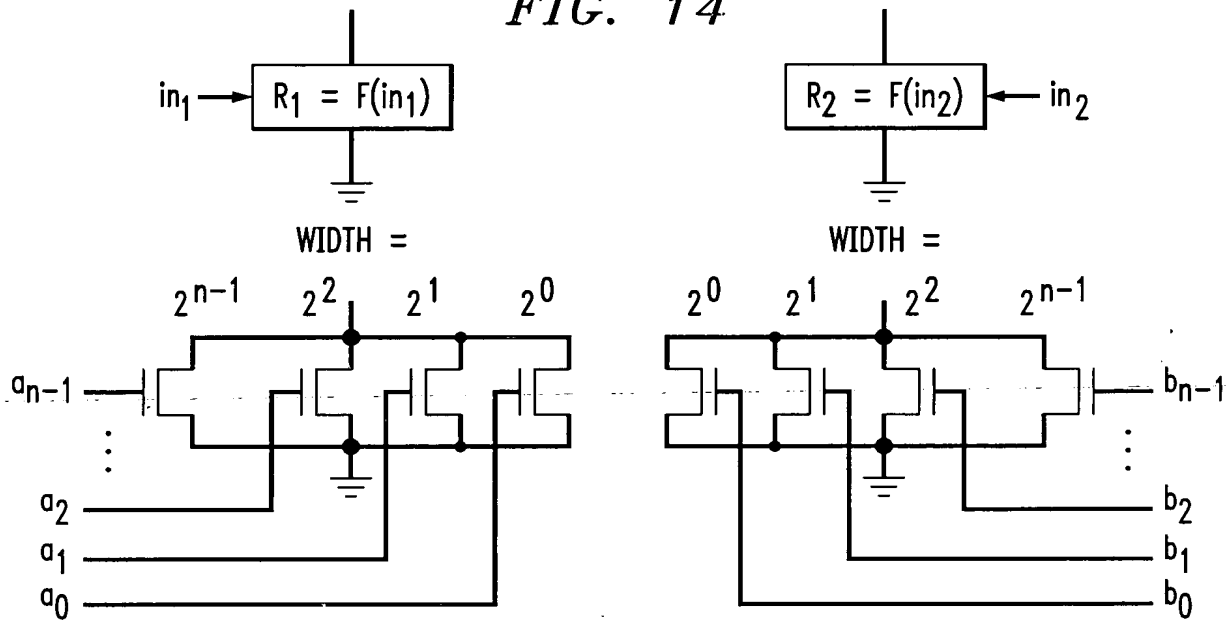


FIG. 15

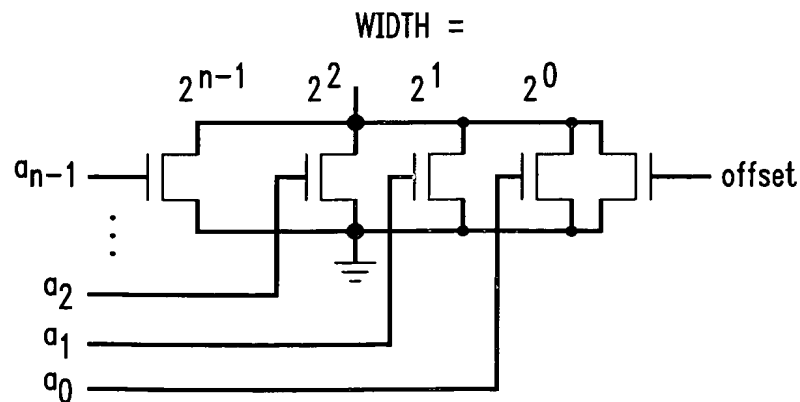
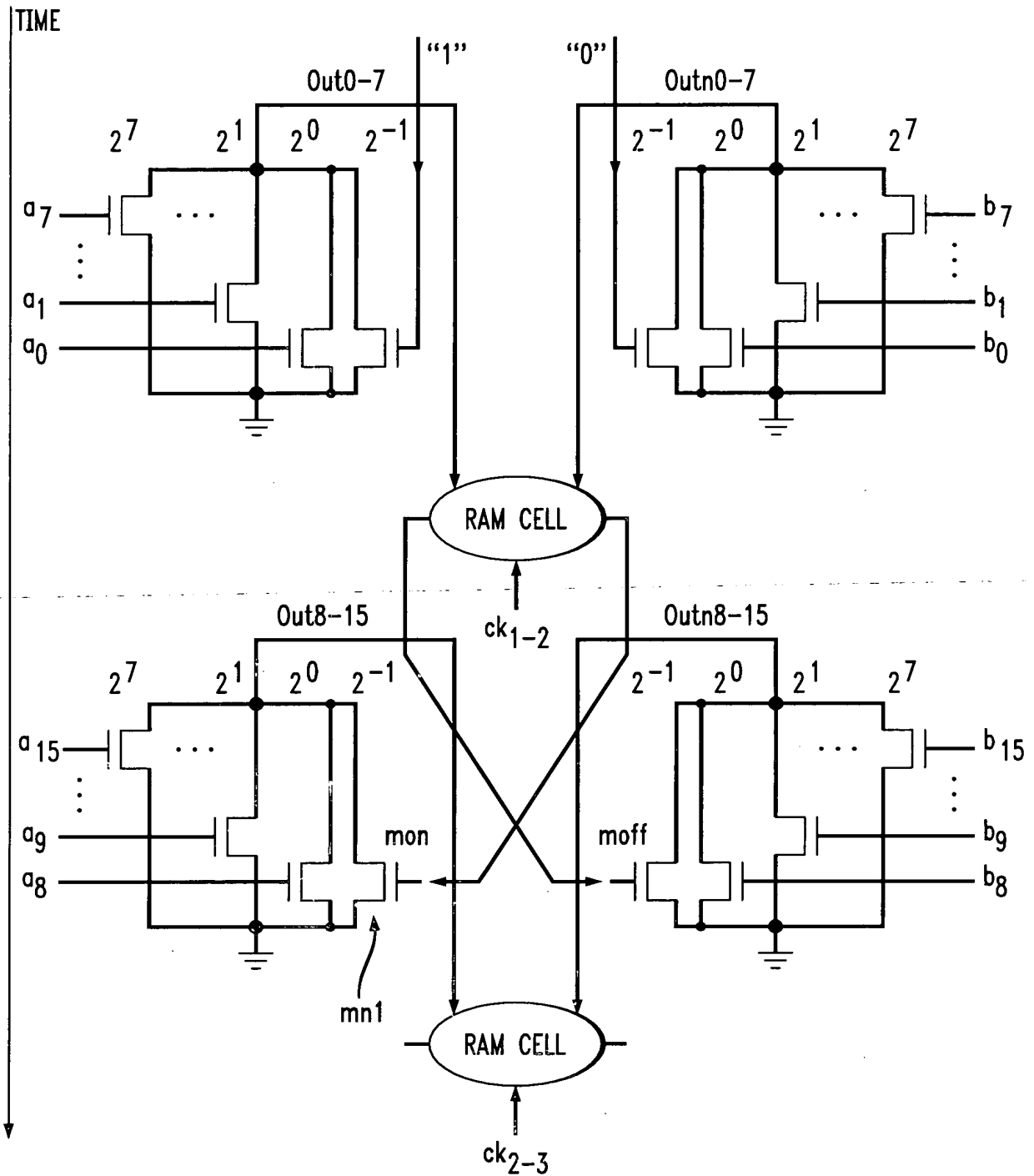
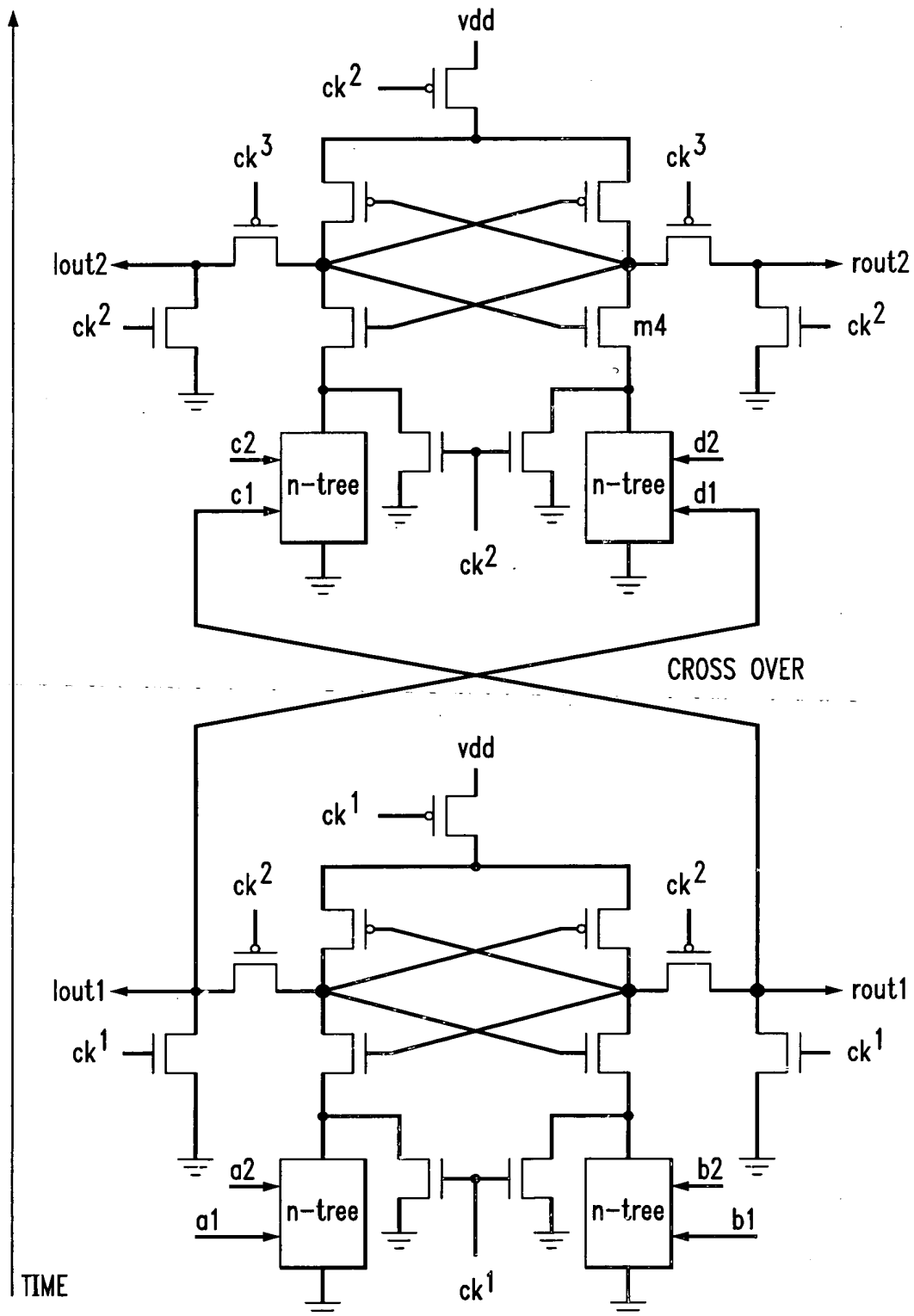


FIG. 16



14/32

FIG. 17



*FIG. 17 (CONT'D)*

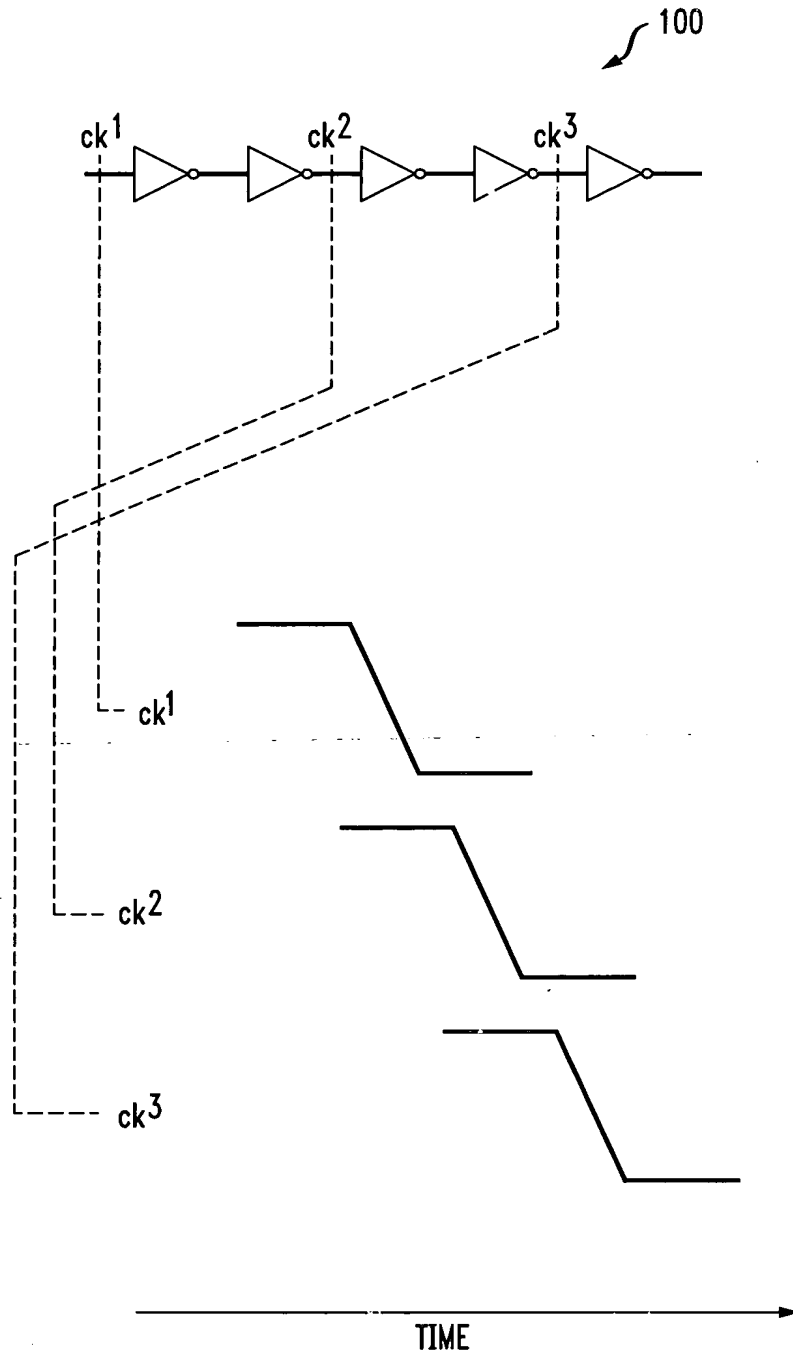
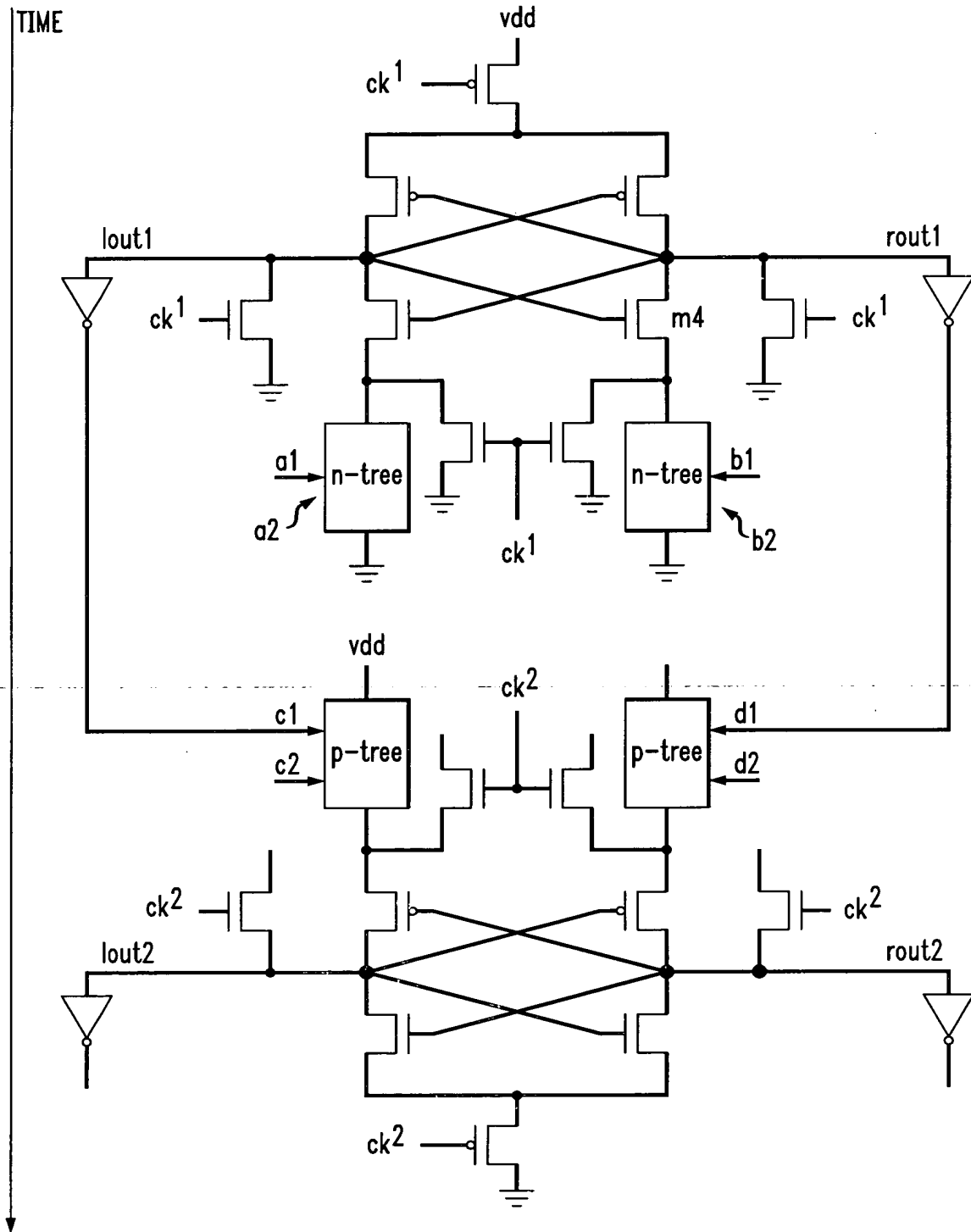
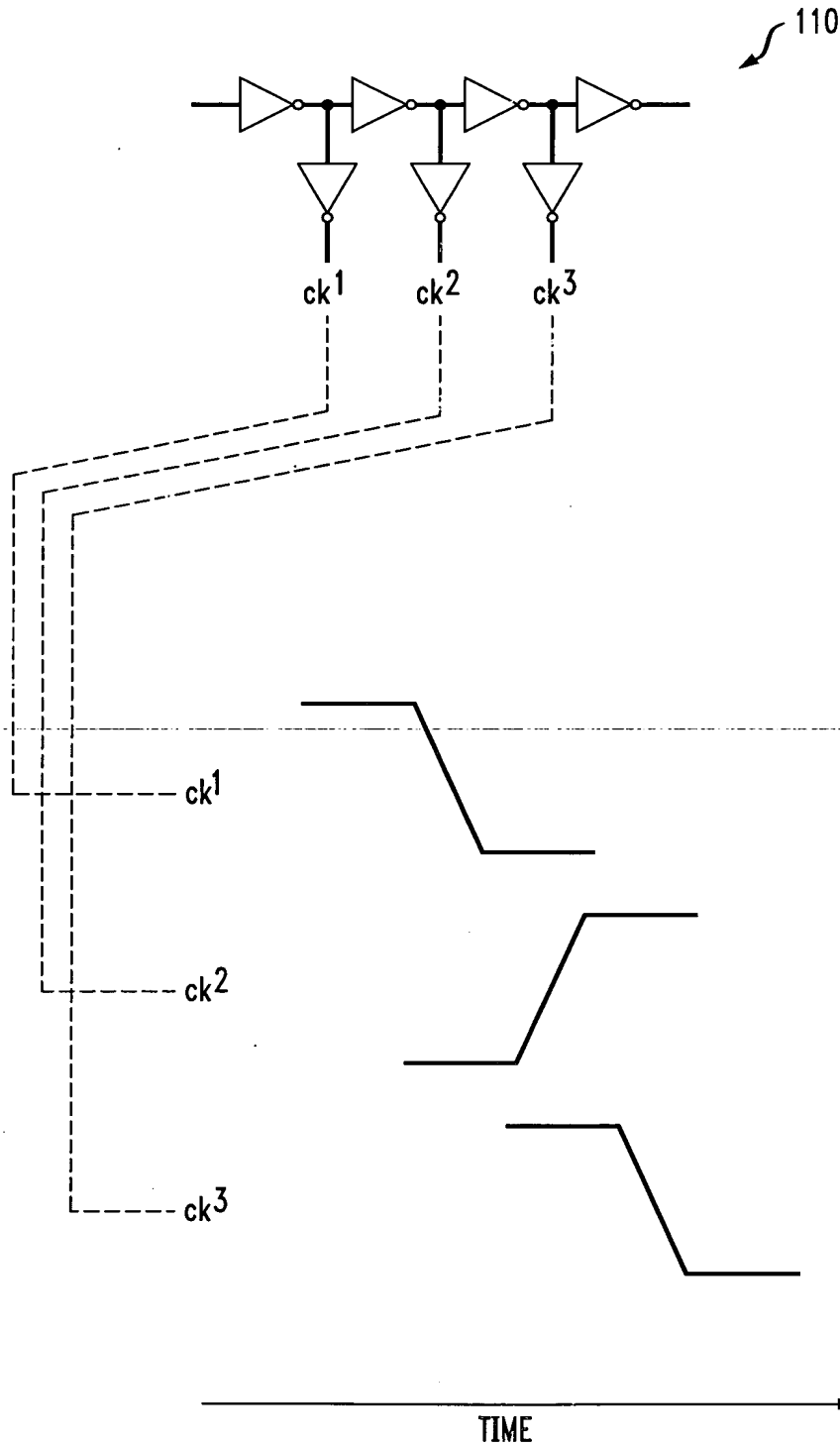


FIG. 18





*FIG. 18 (CONT'D)*



18/32

FIG. 19

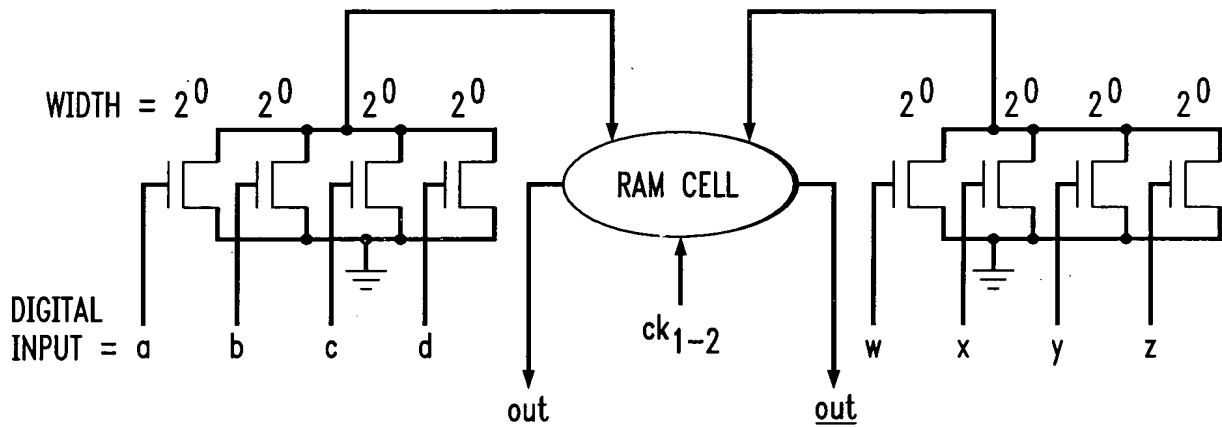


FIG. 20

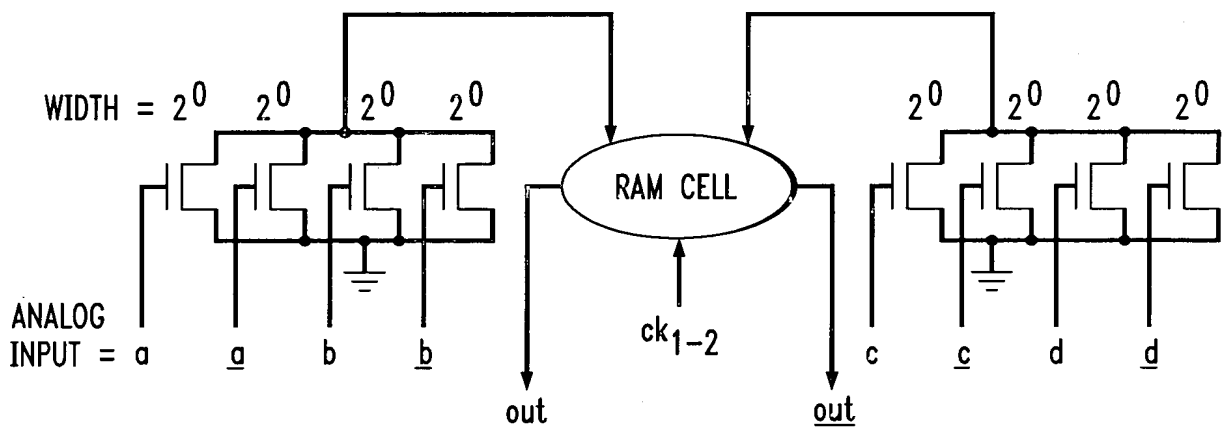


FIG. 21

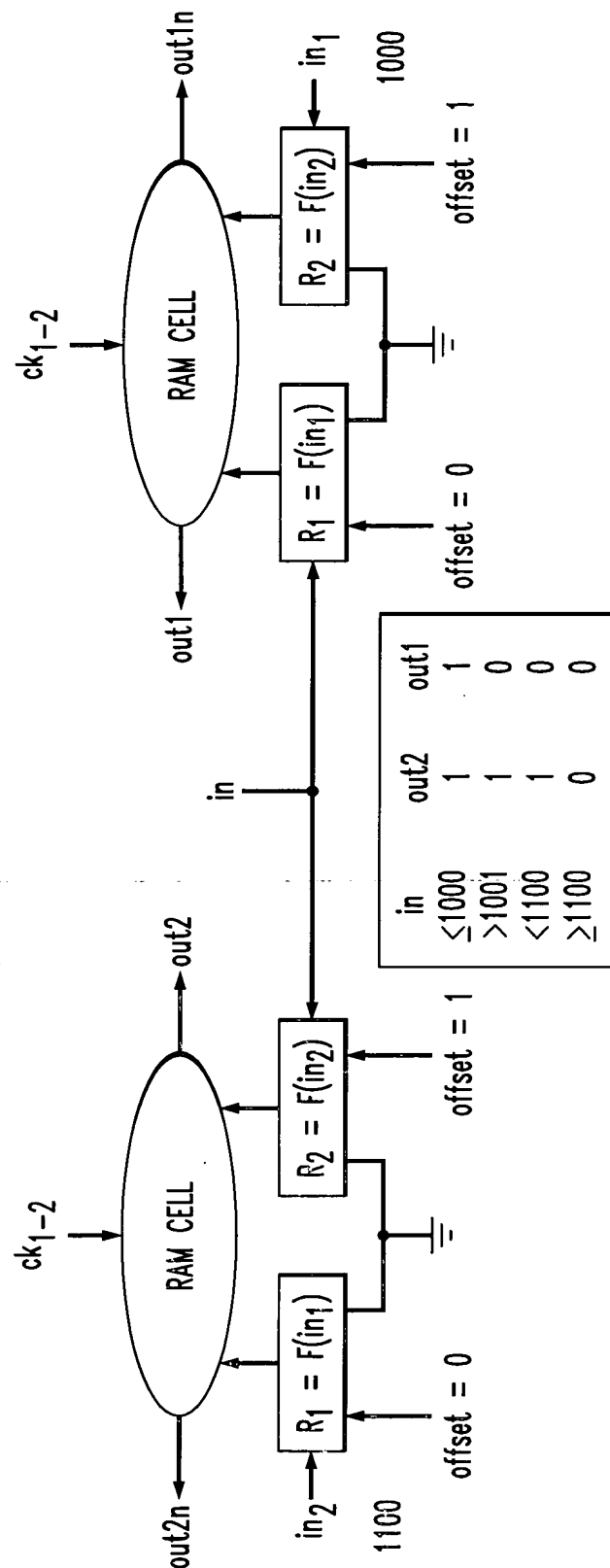
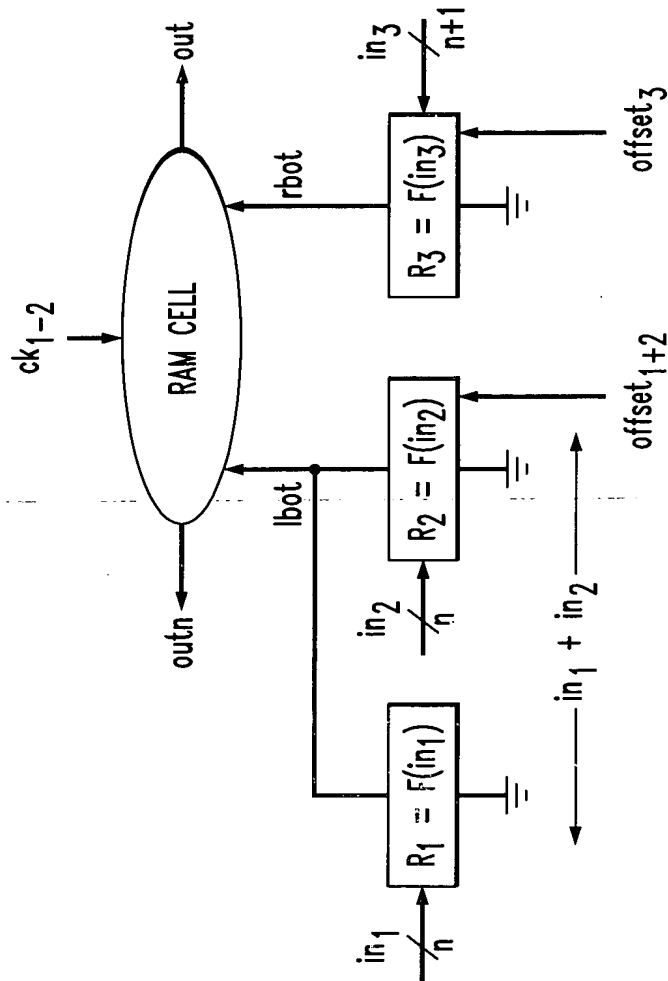


FIG. 22



21/32

FIG. 23

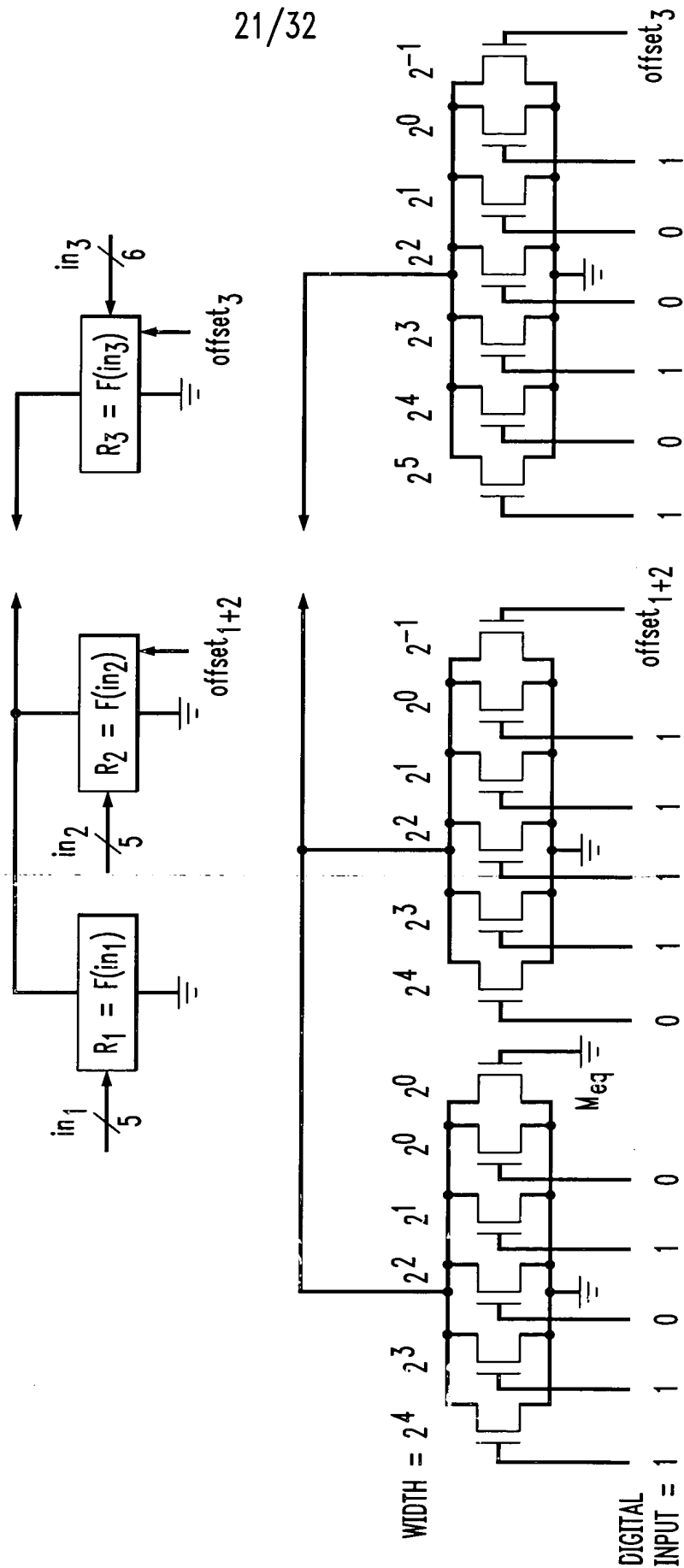
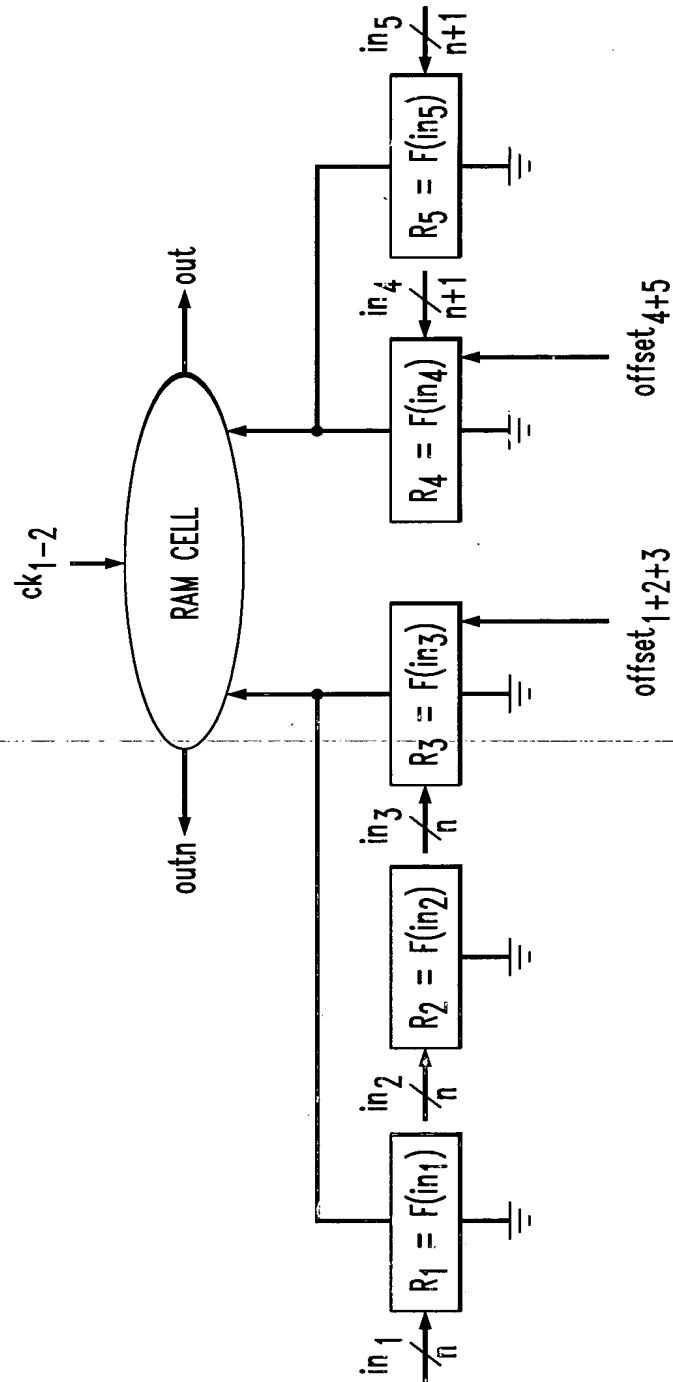


FIG. 24A



23/32

FIG. 24B

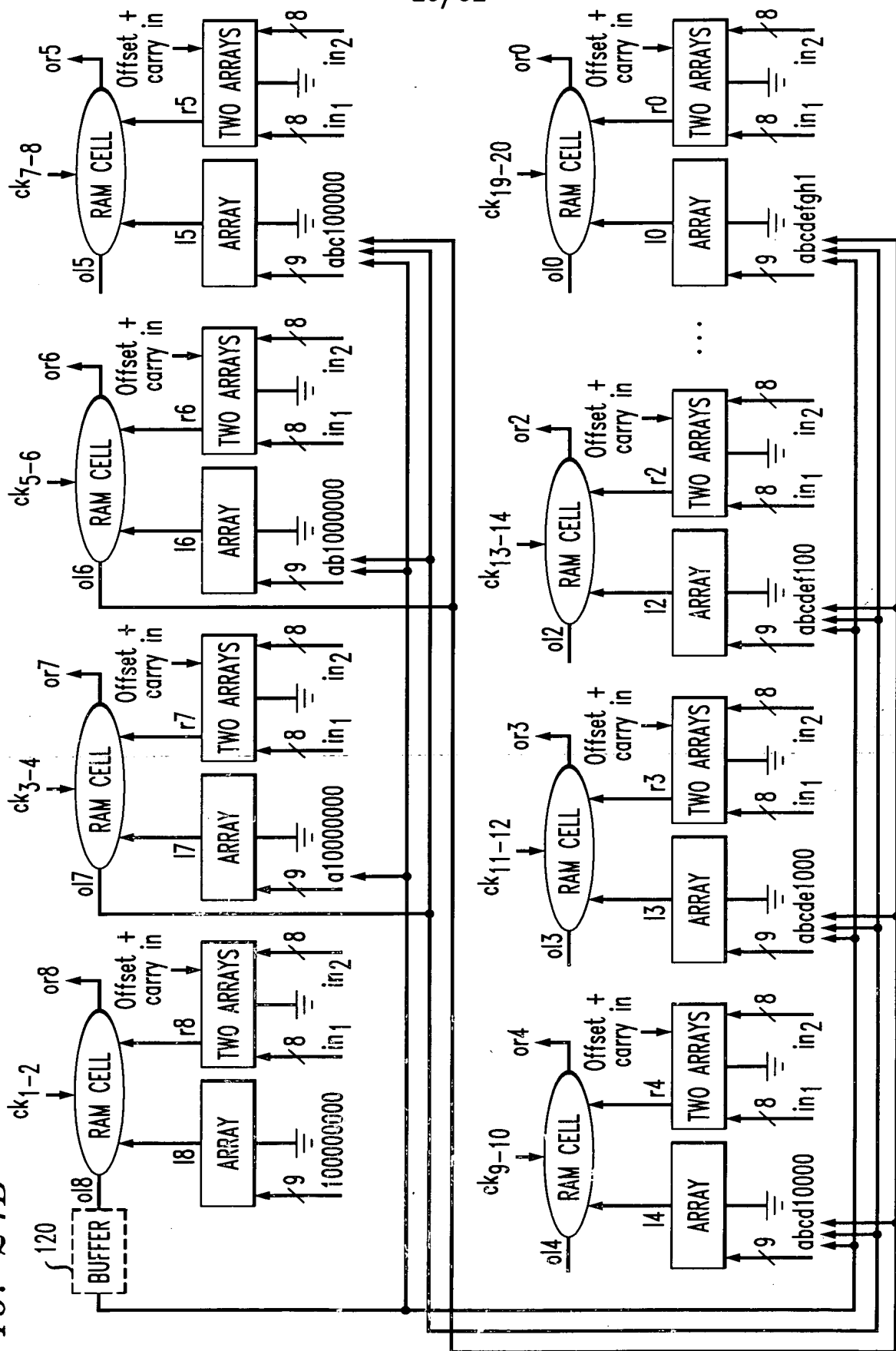
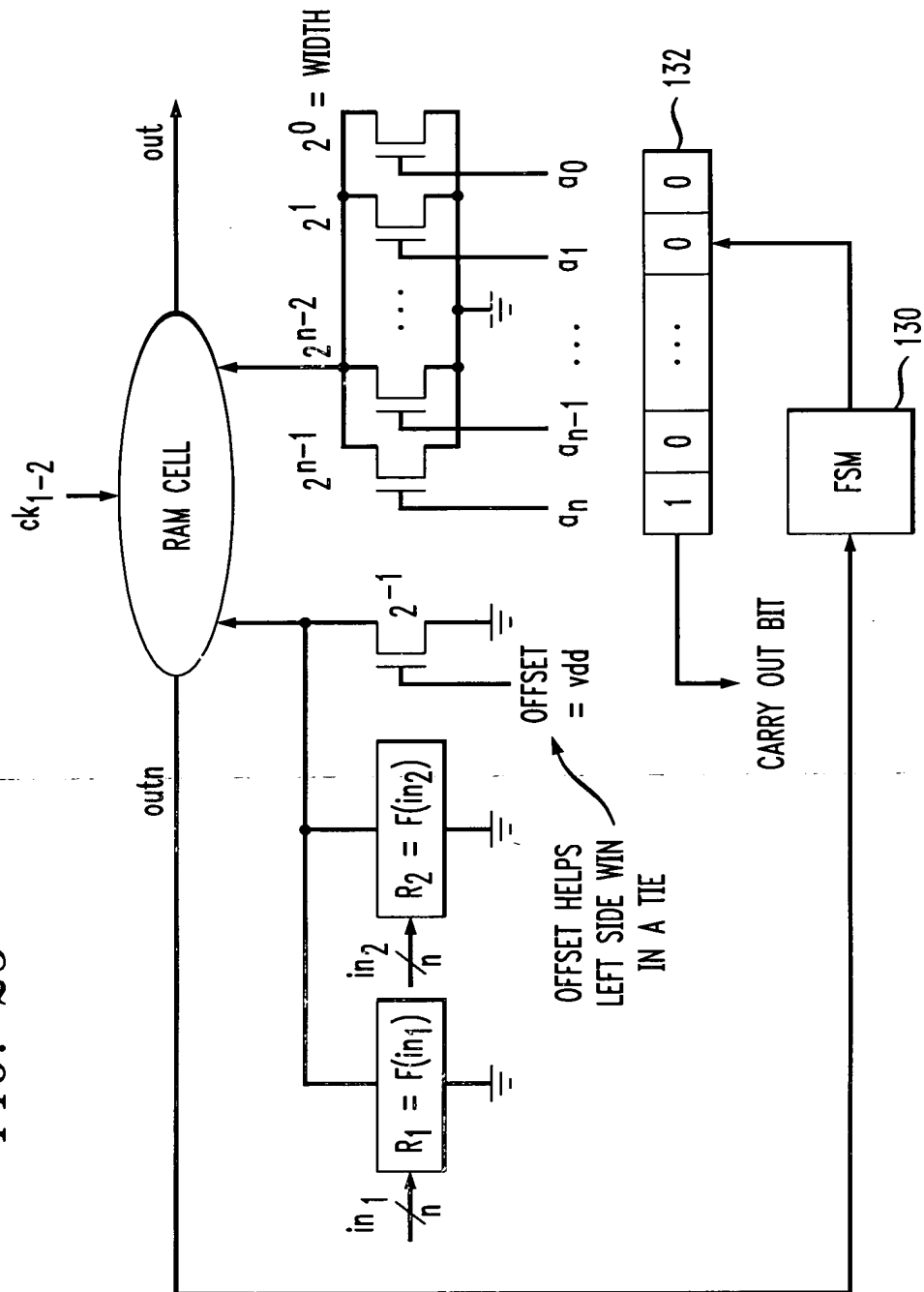


FIG. 25





25/32

FIG. 26

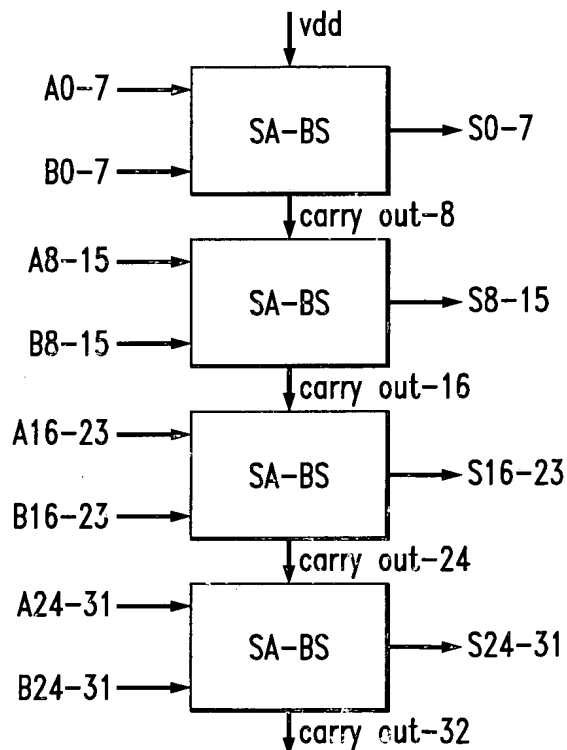
```

/* initialize */
n=8 ;
a[n] = 1;
for (k=1; k<n; k++)
    a[k]=0;

for (i=0; i<n; ++i) /* i=clock tick*/
    if (out = 0)
        a[n-i-1] = 1;
    else {
        a[n-i] = 0;
        if (i < n - 1)
            a[n-i-1] = 1;
    }
/* addition result is in array a[n] */

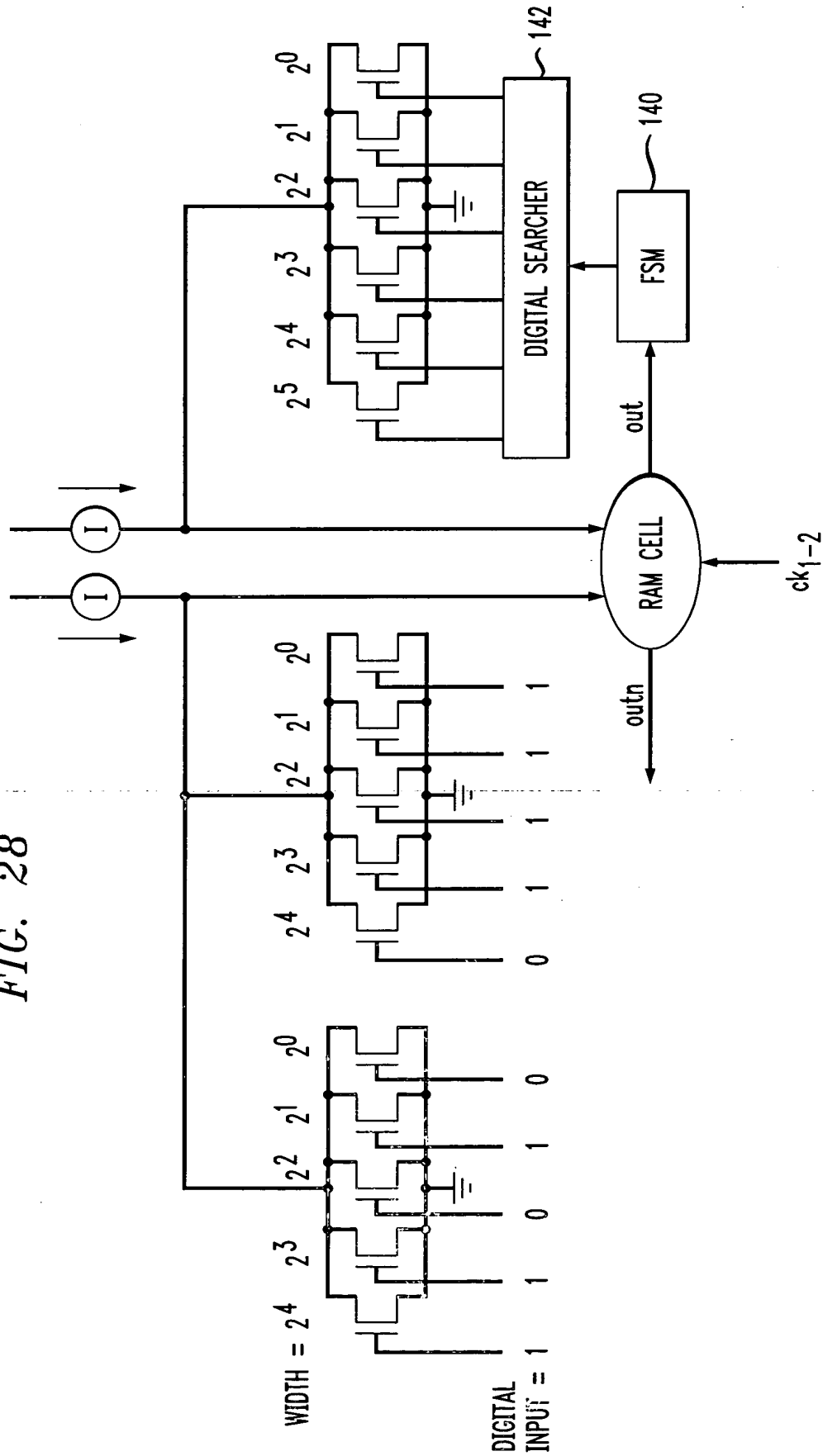
```

FIG. 27



26/32

FIG. 28



27/32

FIG. 29

PRIOR ART

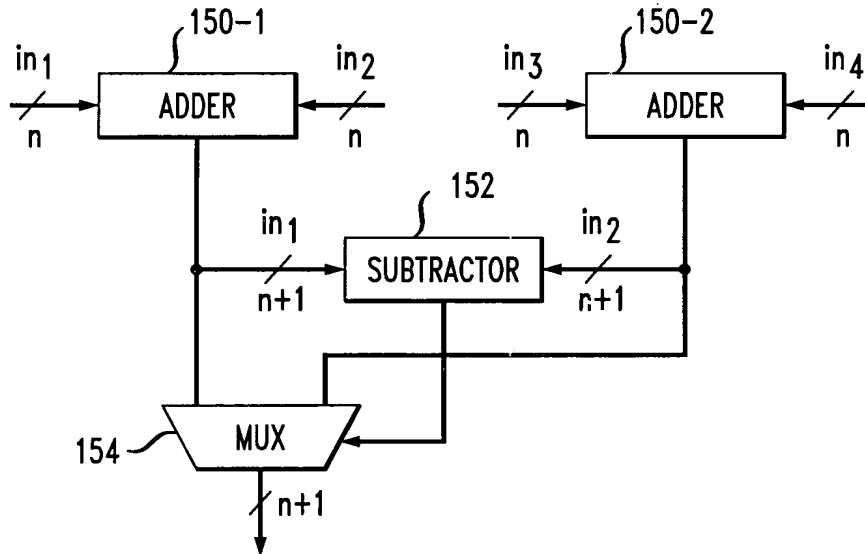


FIG. 30

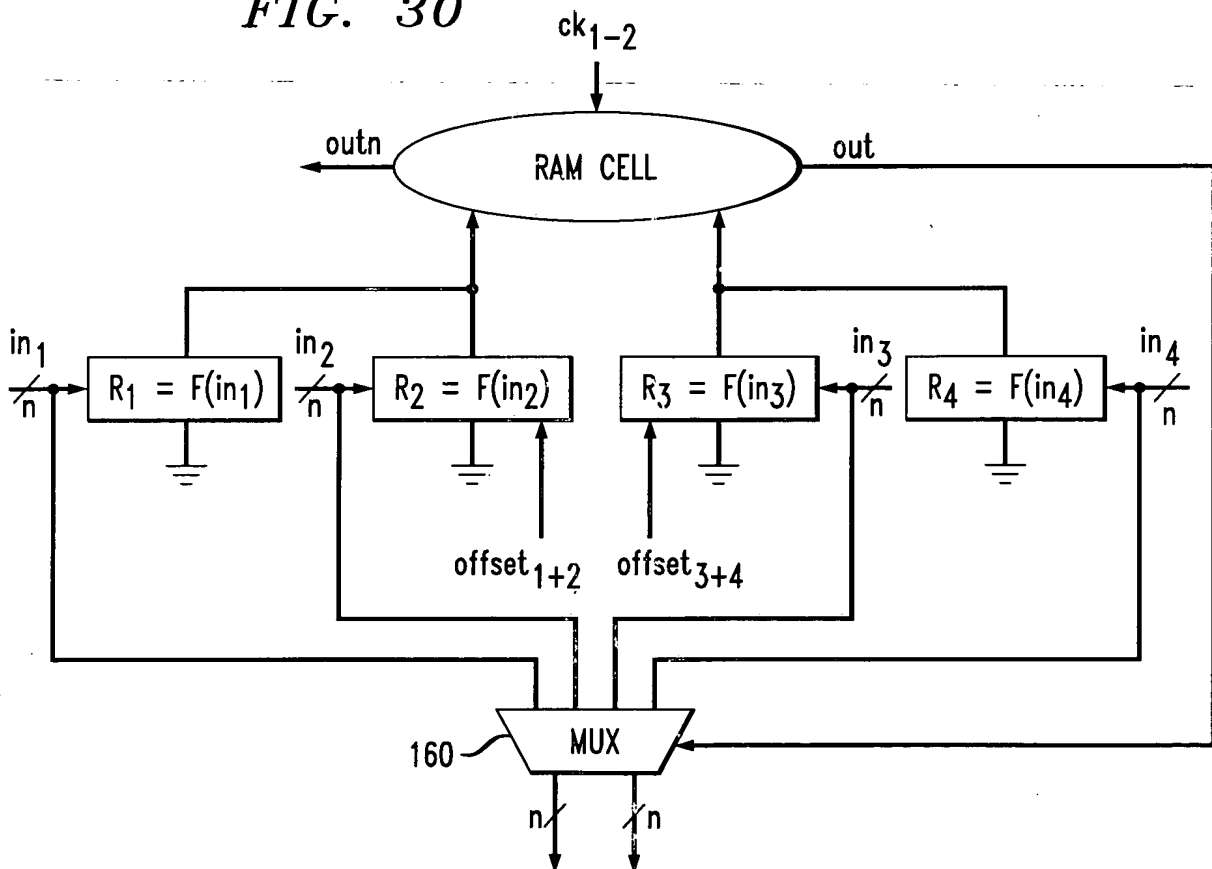


FIG. 31

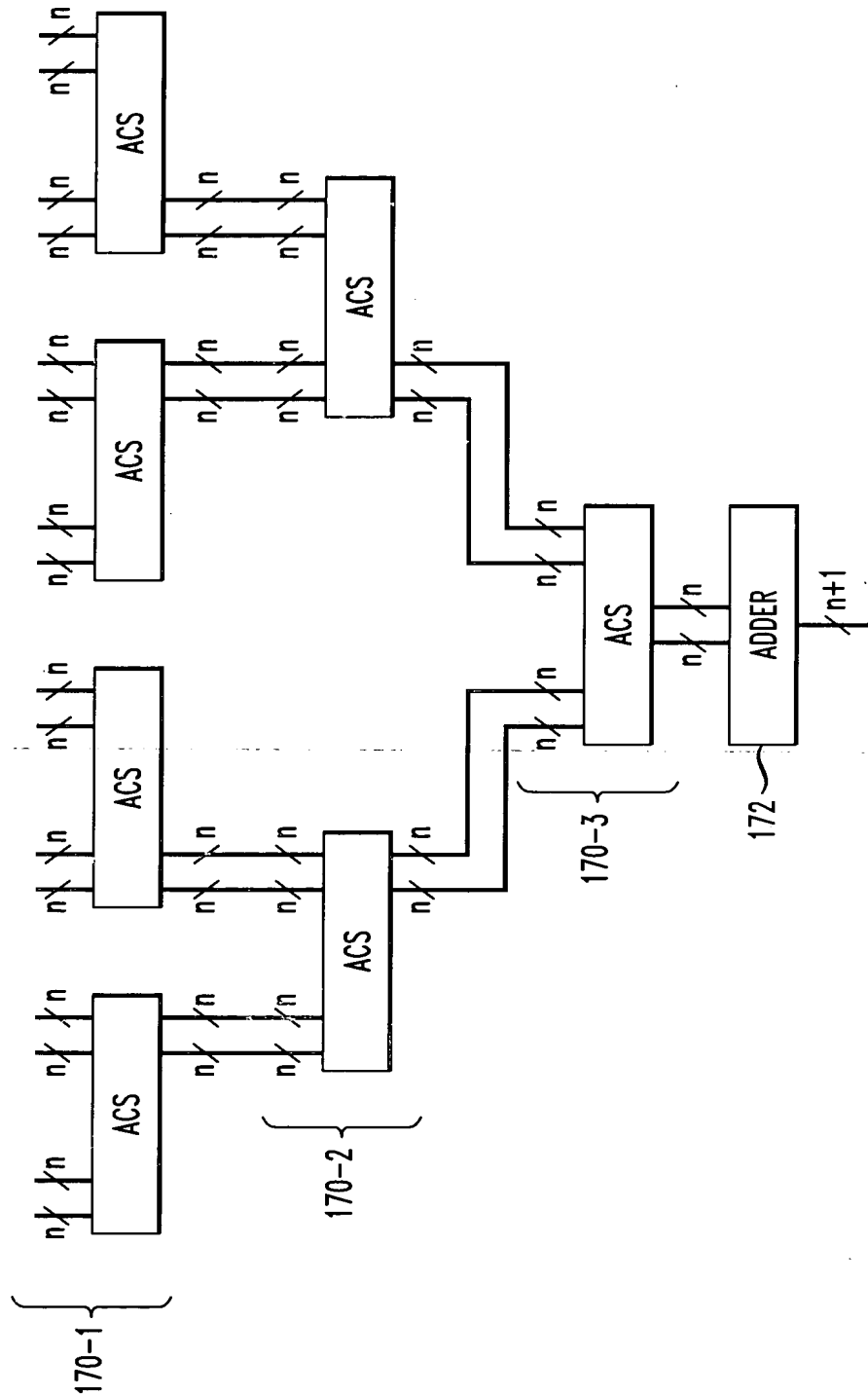


FIG. 32

	POWER ( $\mu$ W)		TRANSISTOR COUNT		DELAY (pSEC)	
	ADD COMPARE	SELECT	ADD COMPARE	SELECT	ADD COMPARE	SELECT
CONVENTIONAL	2000	200	5215	58	1120	200
SEESAW	100	300	55	108	516	173
GAIN FACTOR	X20	X0.66	X95	X0.54	X2.17	X1.15

FIG. 33

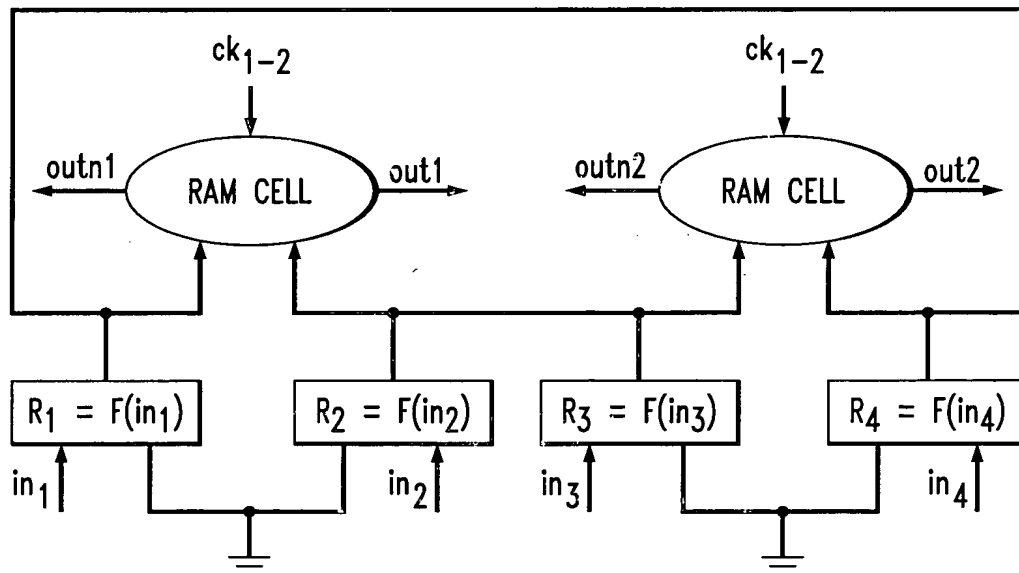


FIG. 34

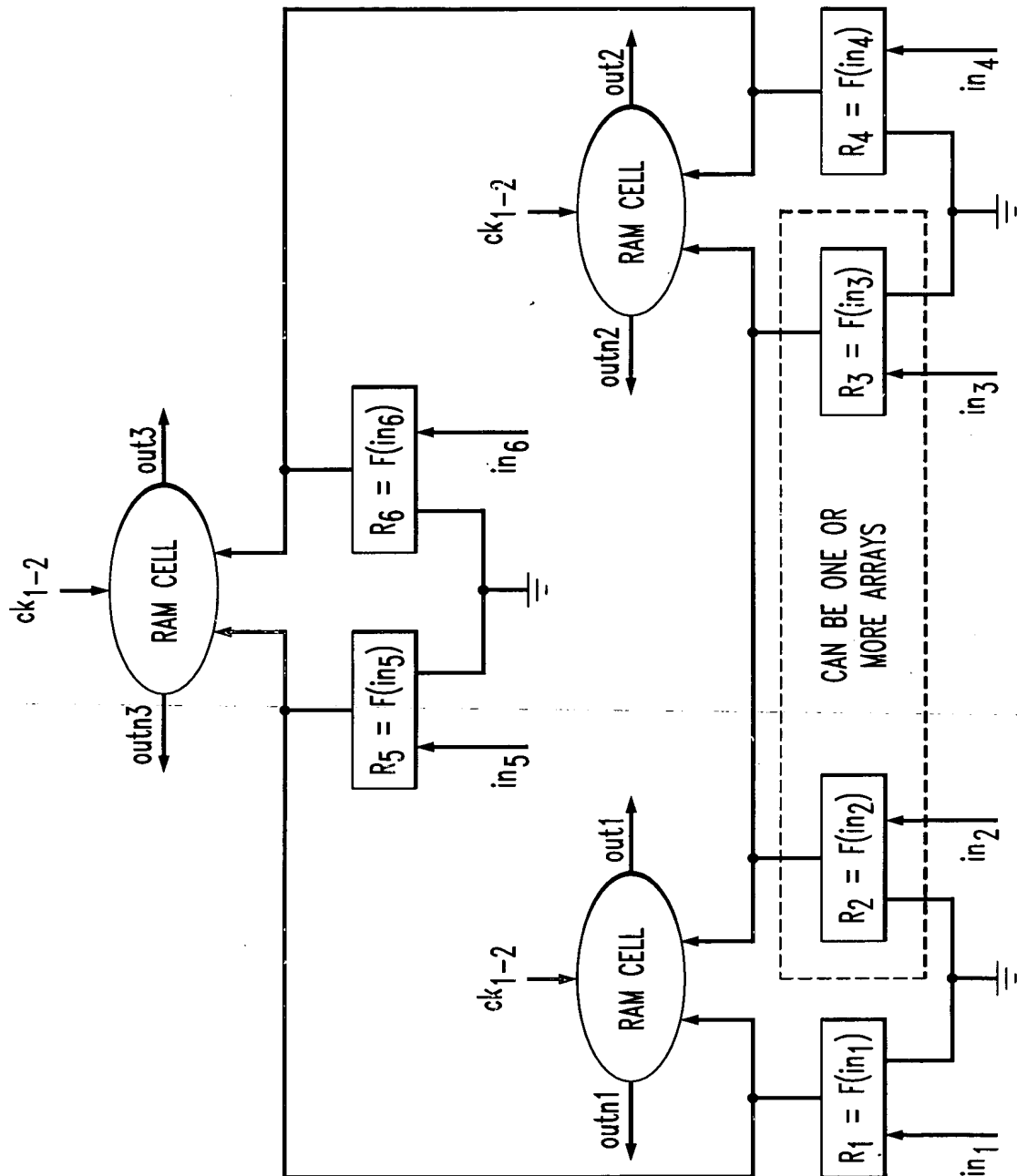


FIG. 35

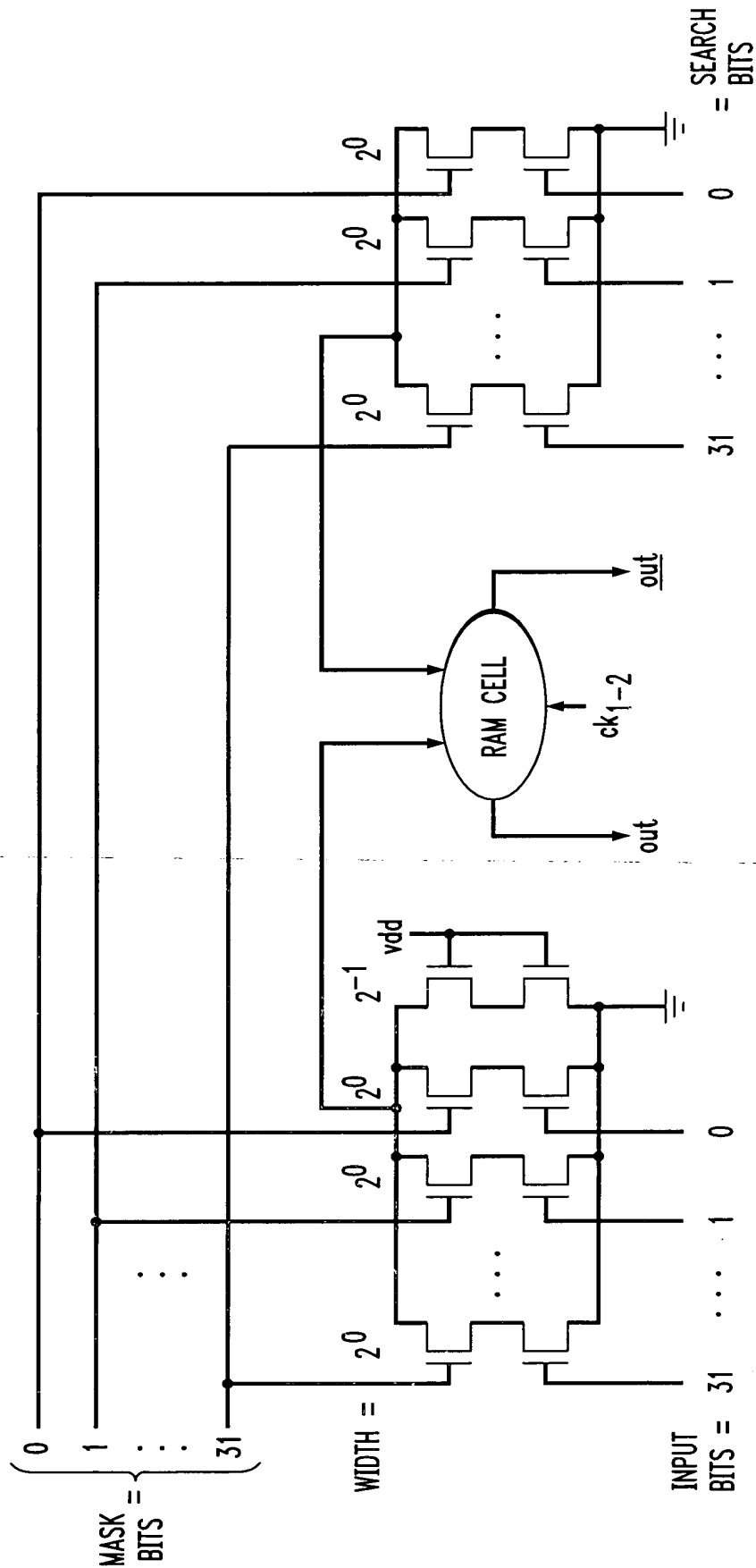


FIG. 36

